XtalOpt User Guide

Release r7.0

Generated by Doxygen 1.7.4

Thu Jun 2 2011 12:22:36

See http://xtalopt.openmolecules.net for the latest versions of these guides.

Contents

1 User documentation **1**

2 XtalOpt Tutorial **3**

2.1 Contents . . . 3

2.2 Launch XtalOpt . . . 4

2.3 Enter composition and restraints . . . 4

2.4 Optimizer setup . . . 4

2.4.1 VASP . . . 5

2.4.2 GULP . . . 6

2.4.3 PWscf . . . 7

2.4.4 CASTEP . . . 8

2.5 Queue setup . . . 9

2.5.1 Using a remote PBS cluster . . . 9

2.5.2 Using a remote SGE cluster . . . 11

2.5.3 Using a remote SLURM cluster . . . 12

2.5.4 Running optimations locally . . . 14

2.6 What is written to the local directory? . . . 14

2.7 Search Settings . . . 15

2.8 "Begin" . . . 16

2.9 Monitor progress . . . 17

2.9.1 View trends . . . 18

2.10 Further reading . . . 19

3 Saving and Resuming Sessions in XtalOpt **21**

3.1 Contents . . . 21

3.2 How to save your session . . . 22

3.3 How to resume your session 22

4 Optimization Schemes **23**

4.1 Contents 23

4.2 Overview: What are optimimization schemes, and why use them? 24

4.2.1 In a nutshell.. 24

4.2.2 More details 24

4.3 Optimization scheme user interface 25

4.3.1 Optimization step list 25

4.3.2 Add new optimization step 26

4.3.3 Remove current optimization step 26

4.3.4 Select template 26

4.3.5 Template editor 26

4.3.6 Save scheme 26

4.3.7 Resume scheme 26

4.4 How to build an optimization scheme? 26

4.5 How to save an optimization scheme for later? 27

4.6 How to load an optimization scheme? 28

4.7 What is saved? 28

4.8 Suggestions for optimization schemes 28

4.8.1 Crystals (XtalOpt) 28

Chapter 1

User documentation

- XtalOpt Tutorial Demonstrates how to use XtalOpt with the various supported optimizers and queuing systems.
- Optimization Schemes What are Optimization Schemes, and how are they used?
- Saving and Resuming Sessions in XtalOpt How to continue a session that has stopped.

Chapter 2

XtalOpt Tutorial

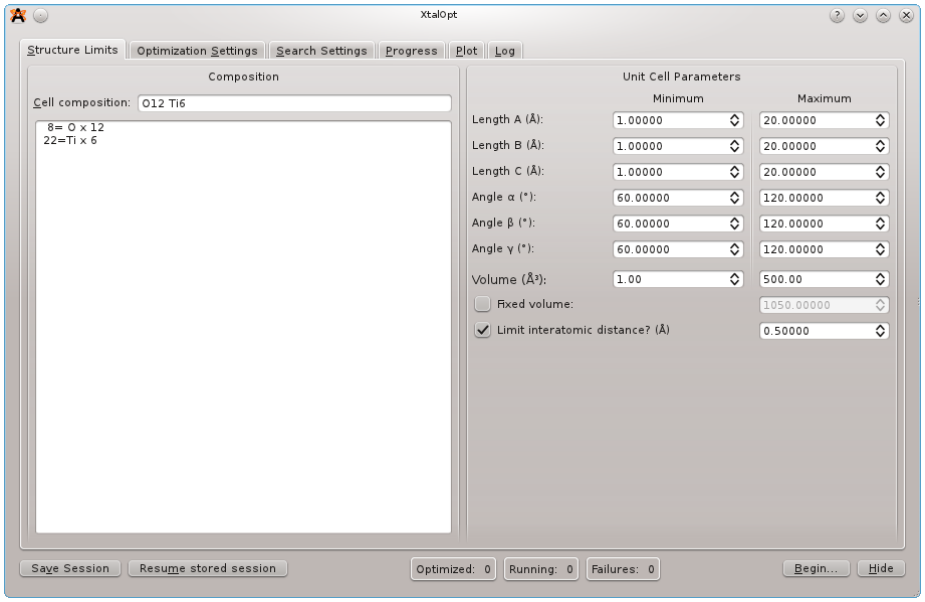
2.1 Contents

- Launch XtalOpt
- Enter composition and restraints
- Optimizer setup
  - VASP
  - GULP
  - PWscf
  - CASTEP
- Queue setup
  - Using a remote PBS cluster
  - Using a remote SGE cluster
  - Using a remote SLURM cluster
  - Running optimations locally
- What is written to the local directory?
- Search Settings
- "Begin"
- Monitor progress
  - View trends
- Further reading
  - Optimization Schemes
  - Saving and Resuming Sessions in XtalOpt

2.2 Launch XtalOpt

Open avogadro, go to the "Extensions" menu and select "XtalOpt".

2.3 Enter composition and restraints

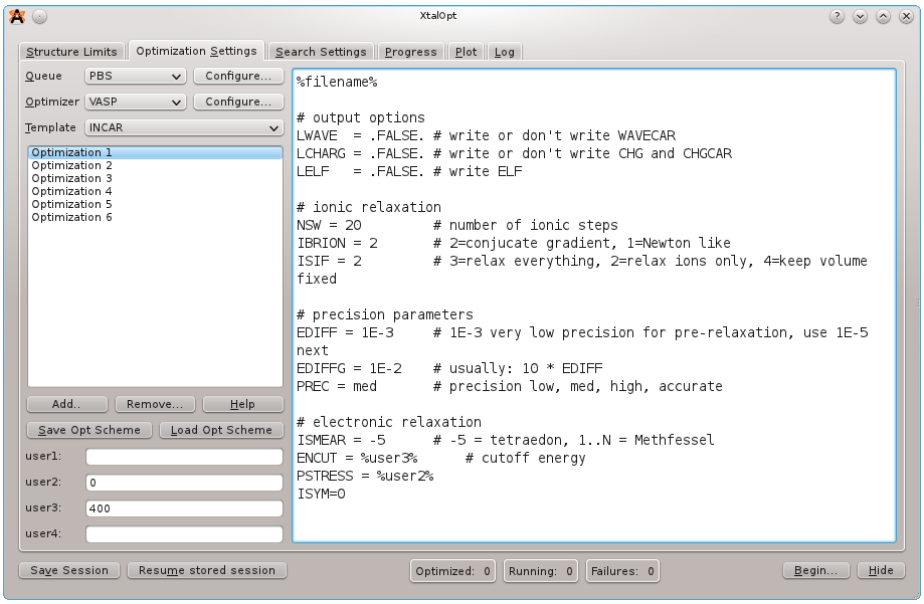


The interface opens to the "Structure Limits" tab, shown above. We will use a 6 formula unit supercell of titanium dioxide for this tutorial, so enter "Ti6 O12" for the cell composition. We will assume that we know nothing about the system and use very loose restraints. Set all cell length minima to 1 angstrom and maxima to 20 angstrom. Constrain the angles to be between 60 and 120 degrees, and the volume from between 1 and 500 cubic angstrom. Specify a minimum interatomic distance of 0.5 angstrom. (Note that due to the angle adjustment described in the CPC 2010 publication, 60-120 degrees is the largest range of cell angles that XtalOpt will generate.)

2.4 Optimizer setup

XtalOpt currently supports the VASP, GULP, PWscf, and CASTEP codes for performing geometry optimizations. Each is detailed in its own section below.

2.4.1 VASP



On the next tab, load the optimization scheme by clicking the "Load Opt Scheme" button and selecting the "samples/vasp-xtalopt.scheme" file that is distributed with the source code. If you do not have a copy of the source code, the scheme file can be obtained by clicking the "Original Format" link at the bottom of the page here.

For more details on optimization schemes, see Optimization Schemes.

After loading the optimization scheme, XtalOpt will prompt for the POTCAR files to use. Select files appropriated for the prompted atom. XtalOpt will construct the POTCAR files on the local computer, and then copy them over to the cluster when the calculation is submitted. It is necessary to have the VASP POTCAR files for each atomic species located somewhere on the local computer. See the VASP manual for information on obtaining the POTCAR files.

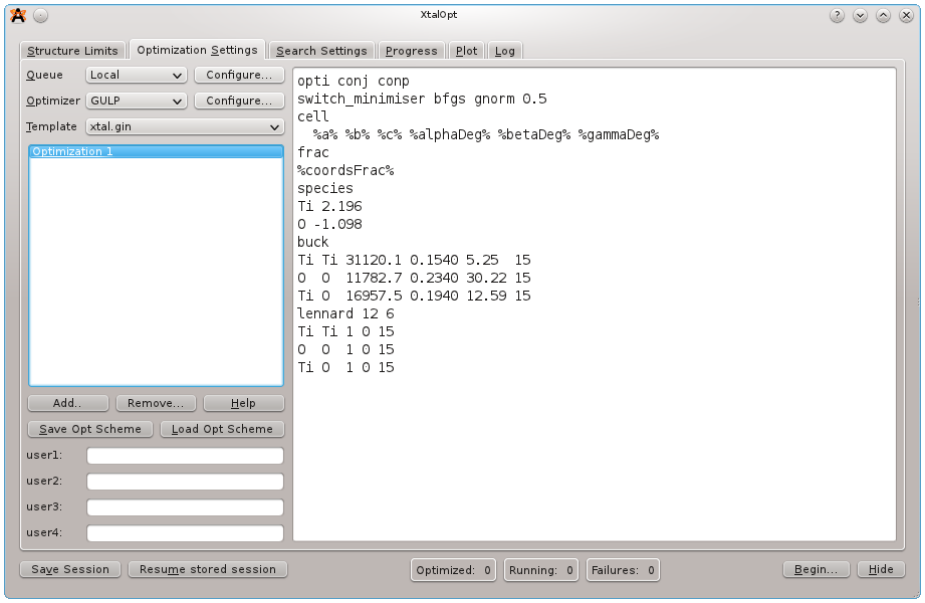
Take a moment to look through each file for each optimization step. Notice that the INCAR template includes two user-specified values, %user2% and %user3% for the external pressure and the energy cutoff, respectively. By entering appropriate values in the "user2:" and "user3:" fields on the left, it is easy to update these values for all optimization steps.

Notice the other %keyword% values in the job.pbs templates. These are used to enter information that is specific to a search or structure when the actual input files are written prior to job submission. Click the "Help" button for a full listing of the available keywords.

XtalOpt expects VASP to use the default filenames, mainly POSCAR, CONTCAR, and OUTCAR.

Skip to next section.

2.4.2 GULP



On the next tab we choose GULP for the local optimizer and enter a template for GULP to use. Select "GULP" as the "Optimizer" and "xtal.gin" as "Template". Next, fill out the text field on the right with the following template:

```
opti conj conp
switch_minimiser bfgs gnorm 0.5
cell
  %a% %b% %c% %alphaDeg% %betaDeg% %gammaDeg%
frac
%coordsFrac%
species
Ti 2.196
O -1.098
buck
Ti Ti 31120.1 0.1540 5.25  15
O  O  11782.7 0.2340 30.22 15
Ti O  16957.5 0.1940 12.59 15
lennard 12 6
Ti Ti 1 0 15
O  O  1 0 15
Ti O  1 0 15
```

Alternatively, one can load the scheme file distributed with the source code under samples/gulp-TiO-xtalopt.scheme. If the source code is not available, the scheme file can be obtained by clicking the "Original Format" link at the bottom of the page here.

For more details on optimization schemes, see Optimization Schemes.

Note the "%" surrounding various keywords. These will be replaced by the structure-specific data when the optimizer is invoked for each structure. Click "Help" to view all of the keywords available. The number of optimization steps can be modified with the "Add/Resume" buttons. The "user" fields in the lower left corner allow users to specify their own keyword/value pairs, which is useful for making changes to multiple

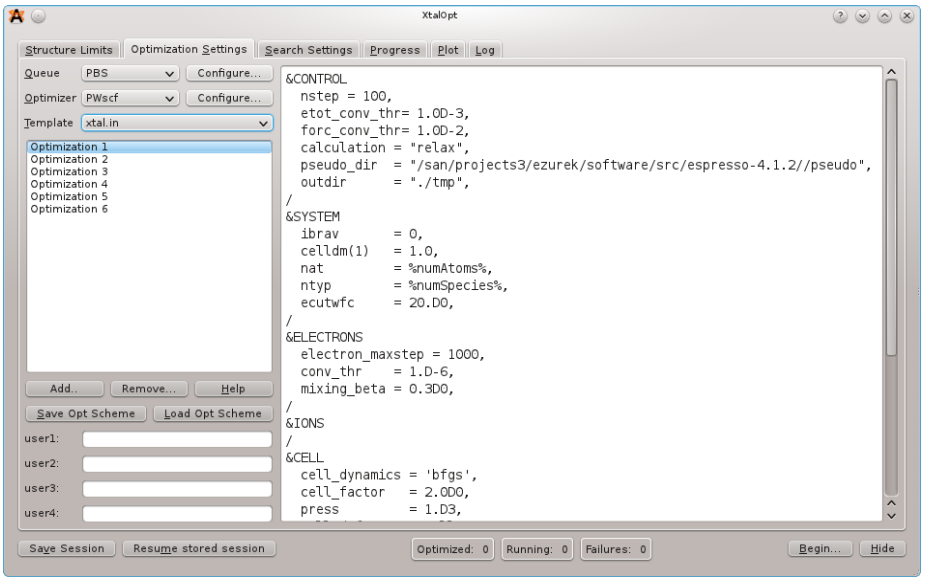
optimization steps at once. We will only be using one optimization step in this tutorial.

XtalOpt expects GULP to use the following filenames:

```
gulp < xtal.gin > xtal.got
```

Skip to next section.

2.4.3 PWscf



On the next tab, load the optimization scheme that is distributed with the source code under the samples/ directory. The scheme that we want is named "pwscf-xtalopt.scheme". If the source code is not available, the scheme file can be obtained by clicking the "Original Format" link at the bottom of the page here.

For more details on optimization schemes, see Optimization Schemes.

Each PWscf input file will need to be edited to specify:

1. The pseudo_dir containing the pseudopotential files on the remote cluster, and
2. The pseudopotentials for each atom (under ATOMIC_SPECIES)

Take a moment to look through each file for each optimization step.

Notice the %keyword% values in the job.pbs templates. These are used to enter information that is specific to a search or structure when the actual input files are written prior to job submission. Click the "Help" button for a full listing of the available keywords.

Be aware that every PWscf/CASTEP installation is different, and it is almost certain that the job.pbs file included with this scheme will not work on any cluster other than the Zurek group's "parity" cluster at SUNY Buffalo's Center for Computational Resources.

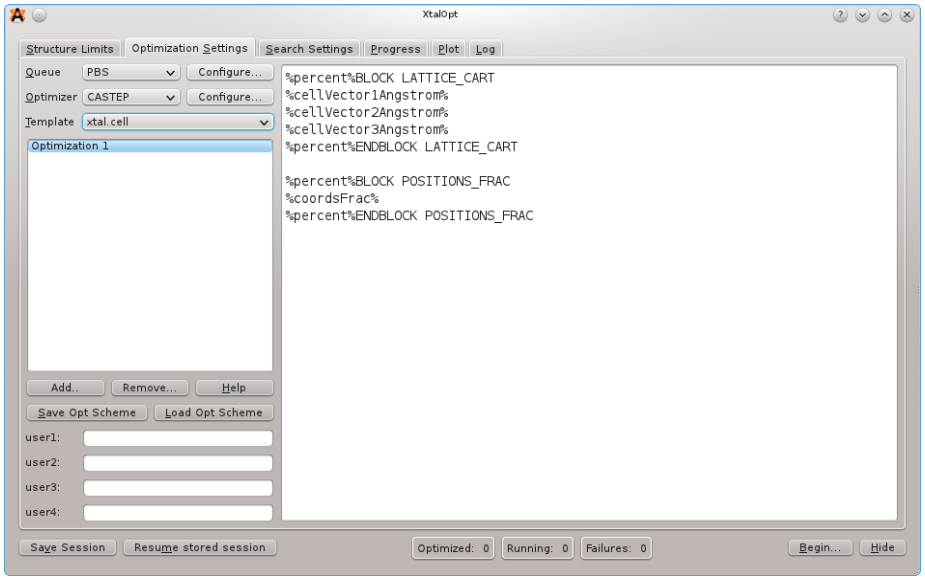
It may take some experimentation to get jobs to submit successfully, and you may need to contact the mantainers of the cluster for assistance for information about MPI, executable locations, etc. Perhaps the easiest method to find the correct PBS script is to run some trial submissions by hand, and then replace the structure/search specific information with the appropriate keywords once a working script has been generated.

XtalOpt expects PWscf to use the following filenames:

```
pw.x < xtal.in > xtal.out
```

Skip to next section.

2.4.4 CASTEP



On the next tab, load the optimization scheme that is distributed with the source code under the samples/ directory. The scheme that we want is named "castep-xtalopt.scheme". If the source code is not available, the scheme file can be obtained by clicking the "Original Format" link at the bottom of the page here.

For more details on optimization schemes, see Optimization Schemes.

It is important to note that CASTEP input files require the "%" character to define blocks. The percent character is special in the XtalOpt input template parser to define keywords (see below). To insert a literal "%" into the input, use percent%.

E.g. Specification of the fractional coordinate block in the .cell template should look like:

```
%percent%BLOCK POSITIONS_FRAC
%coordsFrac%
%percent%ENDBLOCK POSITIONS_FRAC
```

Take a moment to look through each file for each optimization step.

Notice the %keyword% values in the job.pbs templates. These are used to enter information that is specific to a search or structure when the actual input files are written prior to job submission. Click the "Help" button for a full listing of the available keywords.

Be aware that every PWscf/CASTEP installation is different, and it is almost certain that the job.pbs file included with this scheme will not work on any cluster other than the Zurek group's "parity" cluster at SUNY Buffalo's Center for Computational Resources. It may take some experimentation to get jobs to submit successfully, and you may need to contact the mantainers of the cluster for assistance for information about MPI, executable locations, etc. Perhaps the easiest method to find the correct PBS script is to run some trial submissions by hand, and then replace the structure/search specific information with the appropriate keywords once a working script has been generated.

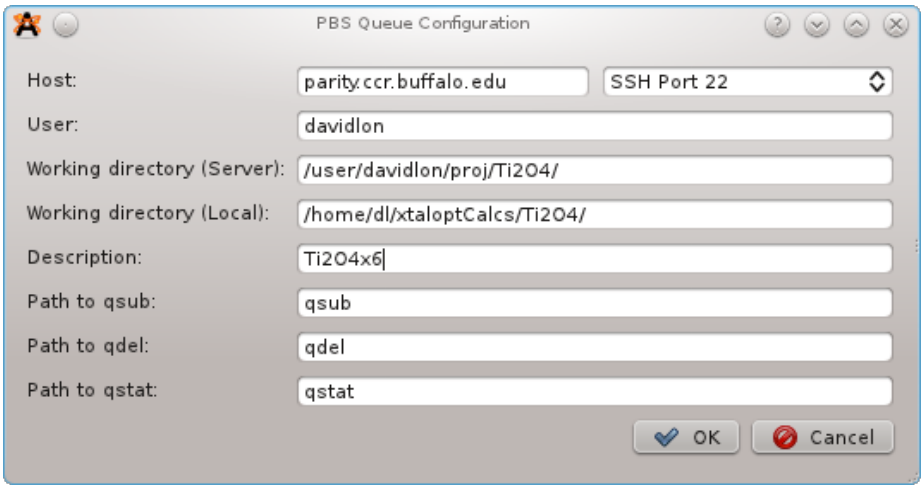
XtalOpt expects CASTEP to use the following filenames:

```
# XtalOpt will write xtal.cell, xtal.param
castep xtal
# CASTEP will create xtal.castep
```

2.5 Queue setup

XtalOpt currently supports using the PBS, SGE, and SLURM queuing systems on remote SSH-accessible clusters, as well as an internal local queue that manages calculations on the user's workstation. Each queueing interface is detailed in its own section below.

2.5.1 Using a remote PBS cluster



Select "PBS" from the list of Queues, and then click the "Configure..." button. A new window will prompt for:

- host: The hostname of the PBS cluster's head node

- user: The username used to log into the cluster
- Working directory (Server): A directory that is readable/writable by "user" on the cluster, used when performing optimizations.
- Working directory (Local): A directory that is readable/writable by the current user on the local computer. This is where the final structures and resume files are written.
- Description: Used for the %description% keyword in input templates.
- Path to qsub: Where to find the qsub executable on the remote cluster. Note that if qsub is in the cluster's $PATH, setting this to just 'qsub' will work.
- Path to qdel: Where to find the qdel executable on the remote cluster. Note that if qdel is in the cluster's $PATH, setting this to just 'qdel' will work.
- Path to qstat: Where to find the qstat executable on the remote cluster. Note that if qstat is in the cluster's $PATH, setting this to just 'qstat' will work.

A new template, "job.pbs" is added to the list of available templates. This is the job submission script for PBS. This script should roughly follow this design:

```
#/bin/bash
#PBS -l nodes=1:ppn=8
#PBS -o ../%gen%x%id%-%optstep%.out
#PBS -e ../%gen%x%id%-%optstep%.err
#PBS -N %description%-%gen%x%id%-%optstep%

###Include this for XtalOpt scripts!###
export PBS_O_WORKDIR=%rempath%

# Change to structure's working directory, copy input files to node's scratch dir
      s:
for node in `cat $PBS_NODEFILE | sort | uniq`; do
rsh $node "cp $PBS_O_WORKDIR/* $PBSTMPDIR/;";
done

# Move to the scratch directory
cd $PBSTMPDIR
echo "running in directory $PBSTMPDIR"

# Set any environment variables needed for the optimizer/MPI here:

# Run optimizer, be sure to use the filenames that XtalOpt expects.
# See the template menu in XtalOpt and the example templates in the
# samples/ directory of the XtalOpt sources.

# Don't forget to clean up after MPI if needed!

// Print files from each node
for node in `cat $PBS_NODEFILE | sort | uniq`; do
echo "$node:"
rsh $node "ls -l $PBSTMPDIR"
done
# Copy back results from master node's scratch directory
cp $PBSTMPDIR/* $PBS_O_WORKDIR/
```

For more details on optimization schemes, see Optimization Schemes.

Be aware that every installation is different, and it is almost certain that the job.pbs file included with this scheme will not work on any cluster other than the Zurek group's "parity" cluster at SUNY Buffalo's Center for Computational Resources. It may take some experimentation to get jobs to submit successfully, and you may need to contact the mantainers of the cluster for assistance or information about MPI, executable locations, etc. Perhaps the easiest method to find the correct PBS script is to run some trial submissions by hand, and then replace the structure/search specific information with the appropriate keywords once a working script has been generated.

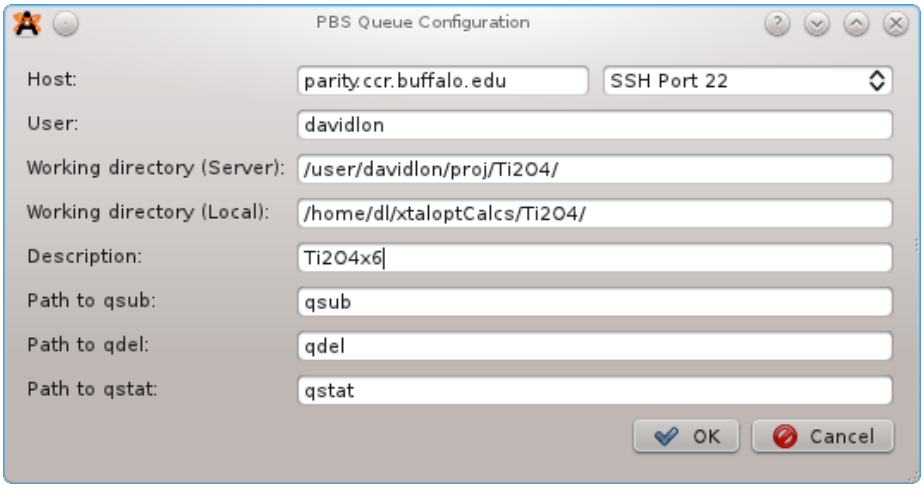
A handy trick for monitoring jobs outside of XtalOpt is to include the following line in job.pbs:

```
#PBS -N %description%-%gen%x%id%-%optstep%
```

This will name each job, for example, xtalSearch-3x4-2, where xtalSearch is a user-specified description of the search, and 3x4-2 means that it is the fourth structure in the third generation running its second optimization step.

Skip to next section.

2.5.2 Using a remote SGE cluster



Todo

Get screenshots of SGE config dialog

Select "SGE" from the list of Queues, and then click the "Configure..." button. A new window will prompt for:

- host: The hostname of the SGE cluster's head node
- user: The username used to log into the cluster

- Working directory (Server): A directory that is readable/writable by "user" on the cluster, used when performing optimizations.
- Working directory (Local): A directory that is readable/writable by the current user on the local computer. This is where the final structures and resume files are written.
- Description: Used for the %description% keyword in input templates.
- Path to qsub: Where to find the qsub executable on the remote cluster. Note that if qsub is in the cluster's $PATH, setting this to just 'qsub' will work.
- Path to qdel: Where to find the qdel executable on the remote cluster. Note that if qdel is in the cluster's $PATH, setting this to just 'qdel' will work.
- Path to qstat: Where to find the qstat executable on the remote cluster. Note that if qstat is in the cluster's $PATH, setting this to just 'qstat' will work.

Todo

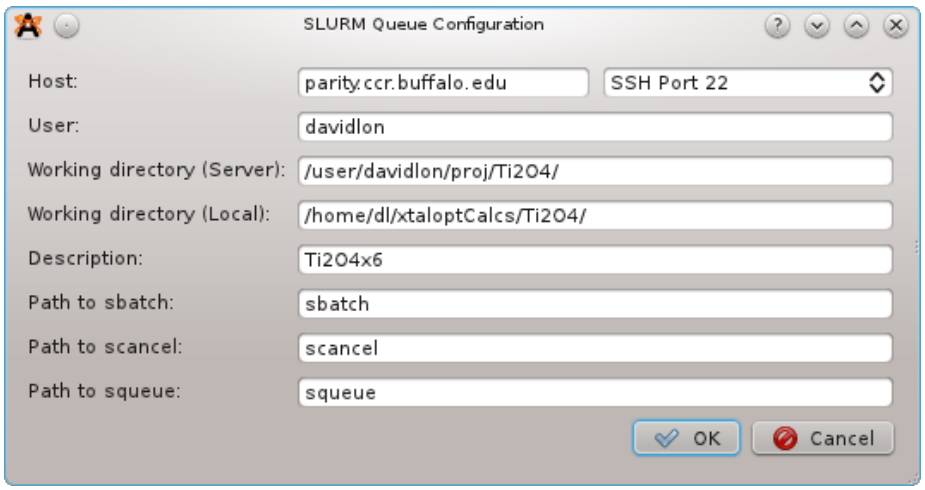
Get template for job.sge scripts

A new template, "job.sge" is added to the list of available templates. This is the job submission script for SGE. It may take some experimentation to get jobs to submit successfully, and you may need to contact the mantainers of the cluster for assistance or information about MPI, executable locations, etc. Perhaps the easiest method to find the correct SGE script is to run some trial submissions by hand, and then replace the structure/search specific information with the appropriate keywords once a working script has been generated.

For more details on optimization schemes, see Optimization Schemes.

Skip to next section.

2.5.3 Using a remote SLURM cluster



Select "SLURM" from the list of Queues, and then click the "Configure..." button. A new window will prompt for:

- host: The hostname of the SGE cluster's head node
- user: The username used to log into the cluster
- Working directory (Server): A directory that is readable/writable by "user" on the cluster, used when performing optimizations.
- Working directory (Local): A directory that is readable/writable by the current user on the local computer. This is where the final structures and resume files are written.
- Description: Used for the %description% keyword in input templates.
- Path to sbatch: Where to find the sbatch executable on the remote cluster. Note that if sbatch is in the cluster's $PATH, setting this to just 'sbatch' will work.
- Path to scancel: Where to find the scancel executable on the remote cluster. Note that if scancel is in the cluster's $PATH, setting this to just 'scancel' will work.
- Path to squeue: Where to find the squeue executable on the remote cluster. Note that if squeue is in the cluster's $PATH, setting this to just 'squeue' will work.

Todo

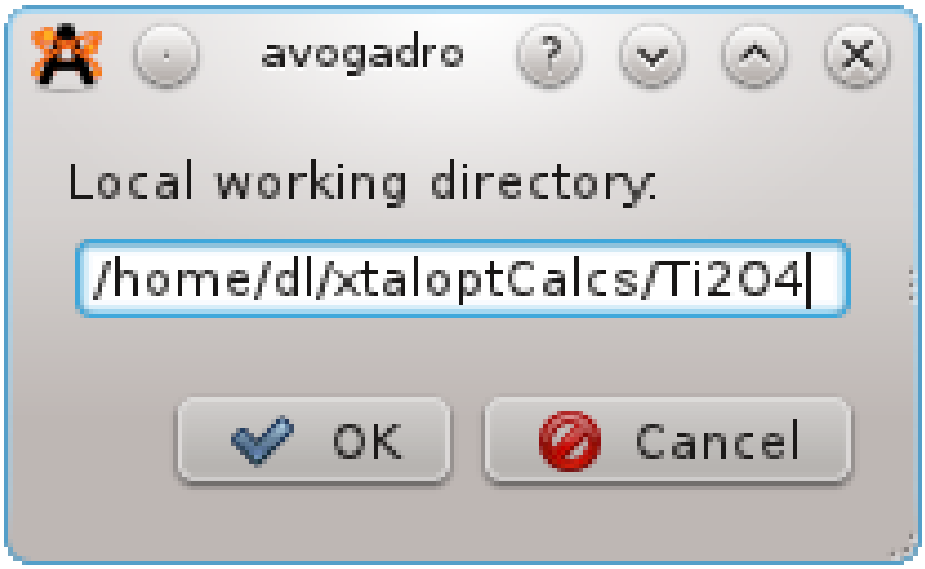
Get template for job.slurm scripts

A new template, "job.slurm" is added to the list of available templates. This is the job submission script for SLURM. It may take some experimentation to get jobs to submit successfully, and you may need to contact the mantainers of the cluster for assistance or information about MPI, executable locations, etc. Perhaps the easiest method to find the correct SLURM script is to run some trial submissions by hand, and then replace the structure/search specific information with the appropriate keywords once a working script has been generated.

For more details on optimization schemes, see Optimization Schemes.

Skip to next section.

2.5.4 Running optimations locally



Select "Local" from the list of Queues, and then click the configure button. A new window will prompt for:

- Local working directory: A directory that is readable/writable by the current user on the local computer. This is where the final structures and resume files are written.

If the optimizer's executable (vasp, gulp, pw.x, castep, etc) is not in your system path, you will need to specify the location of the executable by clicking the "Configure..." button next to the optimizer selection menu.

For more details on optimization schemes, see Optimization Schemes.

2.6 What is written to the local directory?

A directory for each structure is created at

```
[Local working directory]/<gen#>x<id#>
```

that will contain input, output, and data files specific to each structure. Two additional files are also written to the local filesystem:

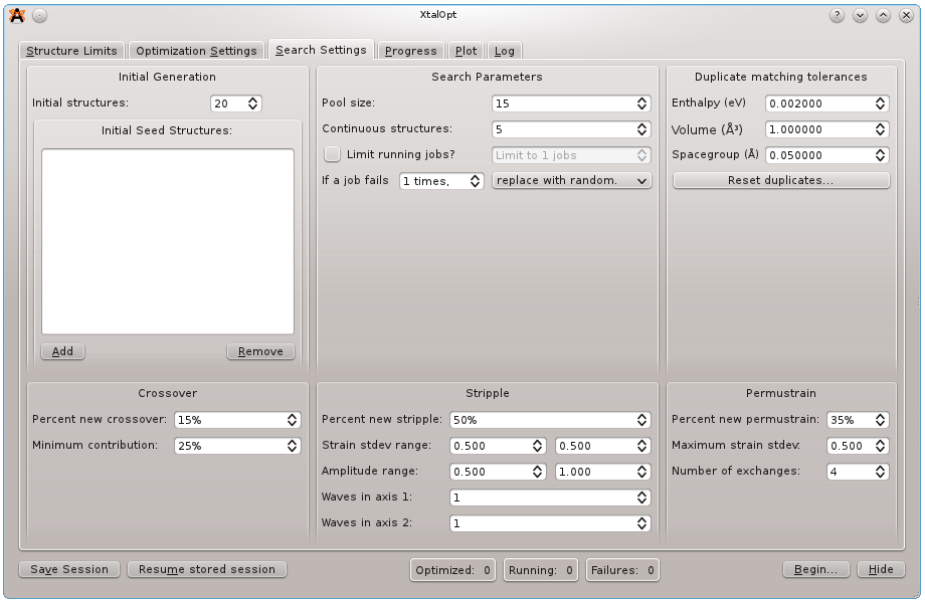
```
[Local working directory]/xtalopt.state
```

which contains save/resume information to continue a session that has been stopped, and

```
[Local working directory]/results.txt
```

which stores a list of all structures sorted by increasing enthalpy. The latter file is handy for offline analysis, since there is no need to open XtalOpt to find the most stable structures of a previous search.

2.7 Search Settings



In the "Search Settings" tab, most of the default settings should suffice (See CPC 2010 publication). We arbitrarily set the initial structures to 20 and the continuous structures to 5, although these may need to be adjusted based on available resources. We will not specify initial seeds, but the option to do so exists on this screen.

It is not neccessary to limit the number of running jobs unless running locally, as the PBS queue on the cluster will manage job control for us. If running locally, set the job limit no higher than [number of available processor cores] - 1 (e.g. for a quadcore processor, allow three jobs to run simultaneously). This allows one core to remain free for the system to run.

2.8 ”Begin”

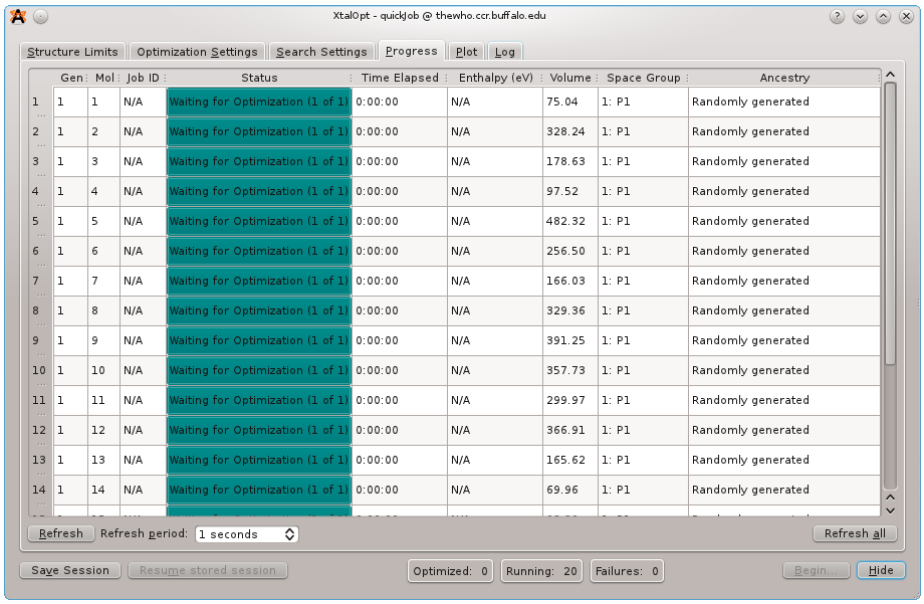

XtalOpt - quickJob @ thewho.ccr.buffalo.edu
Structure Limits
Optimization Settings
Search Settings
Progress
Plot
Log
Gen
Mol
Job ID
Status
Time Elapsed
Enthalpy (eV)
Volume
Space Group
Ancestry
1 1 1 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 75.04 1: P1 Randomly generated
2 1 2 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 328.24 1: P1 Randomly generated
3 1 3 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 178.63 1: P1 Randomly generated
4 1 4 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 97.52 1: P1 Randomly generated
5 1 5 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 482.32 1: P1 Randomly generated
6 1 6 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 256.50 1: P1 Randomly generated
7 1 7 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 166.03 1: P1 Randomly generated
8 1 8 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 329.36 1: P1 Randomly generated
9 1 9 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 391.25 1: P1 Randomly generated
10 1 10 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 357.73 1: P1 Randomly generated
11 1 11 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 299.97 1: P1 Randomly generated
12 1 12 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 366.91 1: P1 Randomly generated
13 1 13 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 165.62 1: P1 Randomly generated
14 1 14 N/A Waiting for Optimization (1 of 1) 0:00:00 N/A 69.96 1: P1 Randomly generated
Refresh
Refresh period:
1 seconds
Refresh all
Save Session
Resume stored session
Optimized: 0
Running: 20
Failures: 0
Begin...
Hide

Figure 2.1: The “Progress” tab immediately after starting a search

XtalOpt has everything it needs to start its search at this point; click the "Begin" button in the lower right corner of the application to tell it to start the search algorithm. A progress bar appears as the random first generation is created. Switch to the "Progress" tab and 20 entries will appear, all with a status of "Waiting for Optimization". Click "Refresh" on this tab to begin the local optimizations. From here, XtalOpt will continue to run without user input, starting new optimizations and generating new structures until it is stopped by the user.

2.9 Monitor progress

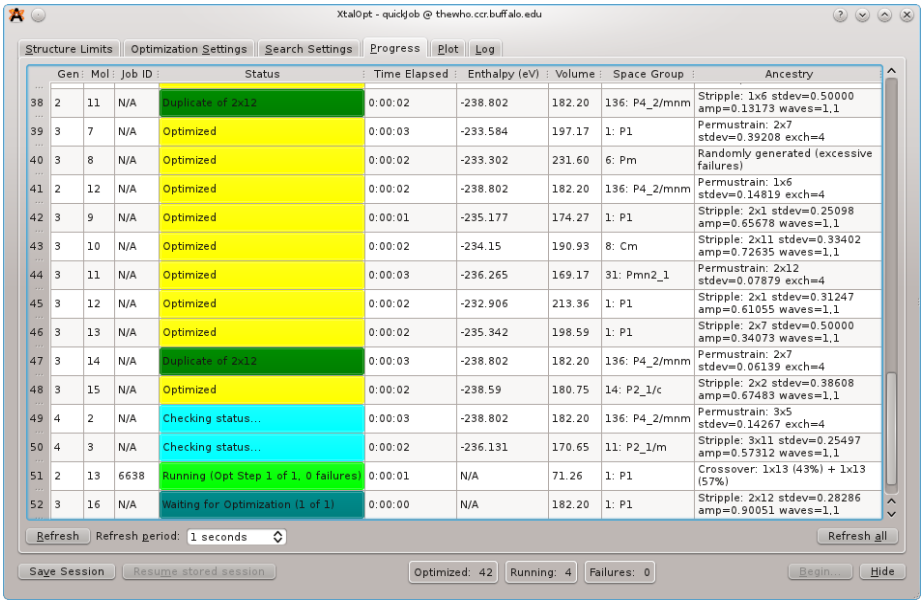

XtalOpt - quickJob @ thewho.ccr.buffalo.edu
Structure Limits
Optimization Settings
Search Settings
Progress
Plot
Log
Gen | Mol | Job ID | Status | Time Elapsed | Enthalpy (eV) | Volume | Space Group | Ancestry
38 | 2 | 11 | N/A | Duplicate of 2x12 | 0:00:02 | -238.802 | 182.20 | 136: P4_2/mnm | Stripple: 1x6 stdev=0.50000 amp=0.13173 waves=1,1
39 | 3 | 7 | N/A | Optimized | 0:00:03 | -233.584 | 197.17 | 1: P1 | Permustrain: 2x7 stdev=0.39208 exch=4
40 | 3 | 8 | N/A | Optimized | 0:00:02 | -233.302 | 231.60 | 6: Pm | Randomly generated (excessive failures)
41 | 2 | 12 | N/A | Optimized | 0:00:02 | -238.802 | 182.20 | 136: P4_2/mnm | Permustrain: 1x6 stdev=0.14819 exch=4
42 | 3 | 9 | N/A | Optimized | 0:00:01 | -235.177 | 174.27 | 1: P1 | Stripple: 2x1 stdev=0.25098 amp=0.65678 waves=1,1
43 | 3 | 10 | N/A | Optimized | 0:00:02 | -234.15 | 190.93 | 8: Cm | Stripple: 2x11 stdev=0.33402 amp=0.72635 waves=1,1
44 | 3 | 11 | N/A | Optimized | 0:00:03 | -236.265 | 169.17 | 31: Pmn2_1 | Permustrain: 2x12 stdev=0.07879 exch=4
45 | 3 | 12 | N/A | Optimized | 0:00:02 | -232.906 | 213.36 | 1: P1 | Stripple: 2x1 stdev=0.31247 amp=0.61055 waves=1,1
46 | 3 | 13 | N/A | Optimized | 0:00:02 | -235.342 | 198.59 | 1: P1 | Stripple: 2x7 stdev=0.50000 amp=0.34073 waves=1,1
47 | 3 | 14 | N/A | Duplicate of 2x12 | 0:00:03 | -238.802 | 182.20 | 136: P4_2/mnm | Permustrain: 2x7 stdev=0.06139 exch=4
48 | 3 | 15 | N/A | Optimized | 0:00:02 | -238.59 | 180.75 | 14: P2_1/c | Stripple: 2x2 stdev=0.38608 amp=0.67483 waves=1,1
49 | 4 | 2 | N/A | Checking status... | 0:00:03 | -238.802 | 182.20 | 136: P4_2/mnm | Permustrain: 3x5 stdev=0.14267 exch=4
50 | 4 | 3 | N/A | Checking status... | 0:00:02 | -236.131 | 170.65 | 11: P2_1/m | Stripple: 3x11 stdev=0.25497 amp=0.57312 waves=1,1
51 | 2 | 13 | 6638 | Running (Opt Step 1 of 1, 0 failures) | 0:00:01 | N/A | 71.26 | 1: P1 | Crossover: 1x13 (43%) + 1x13 (57%)
52 | 3 | 16 | N/A | Waiting for Optimization (1 of 1) | 0:00:00 | N/A | 182.20 | 1: P1 | Stripple: 2x12 stdev=0.28286 amp=0.90051 waves=1,1
Refresh
Refresh period: 1 seconds
Refresh all
Save Session
Resume stored session
Optimized: 42
Running: 4
Failures: 0
Begin...
Hide

Figure 2.2: The “Progress” tab mid-run

As XtalOpt performs the search, the progress table continuously updates, providing information about each structure. We see individuals in various stages of completion: most are optimized (in yellow), structure 2x7 has been automatically marked as a duplicate (dark green) of structure 3x3 and removed from the breeding pool, structure 4x4 is currently undergoing a local optimization (light green), while structure 4x5 is waiting to be optimized (blue).

Other useful information is displayed about each structure, such as the time spent in optimization, the optimized enthalpy, the cell volume, spacegroup, and each structure’s ancestory (i.e. parent(s) and parameters for the genetic operator that generated it). A status bar on the bottom of the window shows the number of structures that are optimized, running, and failing at any given time. This information is visible regardless of which tab is currently being viewed.

An additional feature of the progress table is the ability to immediately visualize any of the individuals in the Avogadro main window -- simply clicking on a row in this table will display the three-dimensional structure in Avogadro, where it can be visualized, modified, or exported. If the user would like to add a bit of "intelligent design" to the evolutionary process, a structure can be modified and then resubmitted using a context (right-click) menu from the progress table. The context menu provides tools to (un)kill a structure, resubmit for local optimization at an arbitrary optimization step, or replace a problematic structure with a new, random individual.

2.9.1 View trends

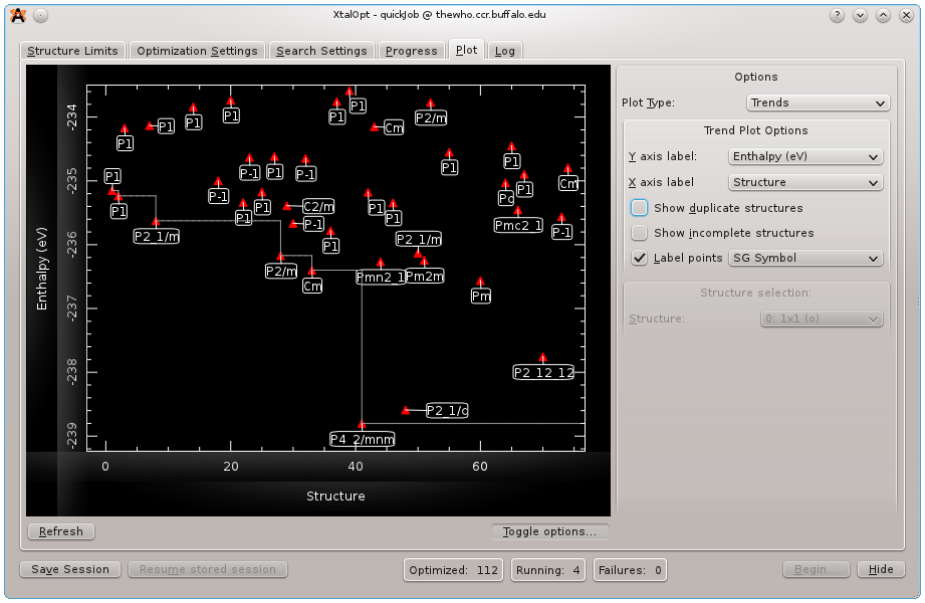


Figure 2.3: The "Plot" tab mid-run displaying enthalpy vs. volume. Each structure is labeled with its Hermann Mauguin spacegroup symbol.

Another visualization and analysis tool available during the search is the interactive plot. The plot is capable of investigating trends in the search by plotting a point for each individual using structure number, generation number, enthalpy, energy, PV enthalpy term, lattice parameters, or cell volume on either axis. This powerful feature allows the user to visualize complex relationships present in the generated structures. E.g., a plot of enthalpy vs. structure number provides an overview of the search's progress. Or, recalling that H = U + PV, plotting enthalpy vs. PV enthalpy term or energy lends insight into whether the enthalpy (H) is dominated by atomic interactions (U) or cell parameters (PV). Further information is available by labeling the points with the individual's spacegroup number, Hermann Mauguin spacegroup symbol, enthalpy, energy, PV term, volume, generation, or index number.

A particularly useful plot is that of enthalpy vs. cell volume, as shown above. From this view, we see a general trend that enthalpy increases with volume (the effect is much more pronounced for systems at higher pressures), and also that below a certain volume enthalpy rises sharply. From this data set, we see that there is a cluster of very low enthalpy structures with cell volumes around 180 cubic angstroms. Armed with this data, we can update the starting volume on the Cell Initialization tab mid-run to reflect this new piece of information that the search has provided us. Many of the other parameters governing structure generation and algorithm specifics can be similarly modified during a search without the need to restart the algorithm.

The plot is also interactive; zooming and panning are possible using simple mouse controls. Clicking on a structure's point on the plot will load it in the main Avogadro

window, allow all the same functionality as described above in Monitor progress.

2.10 Further reading

From here, you may be interested in:

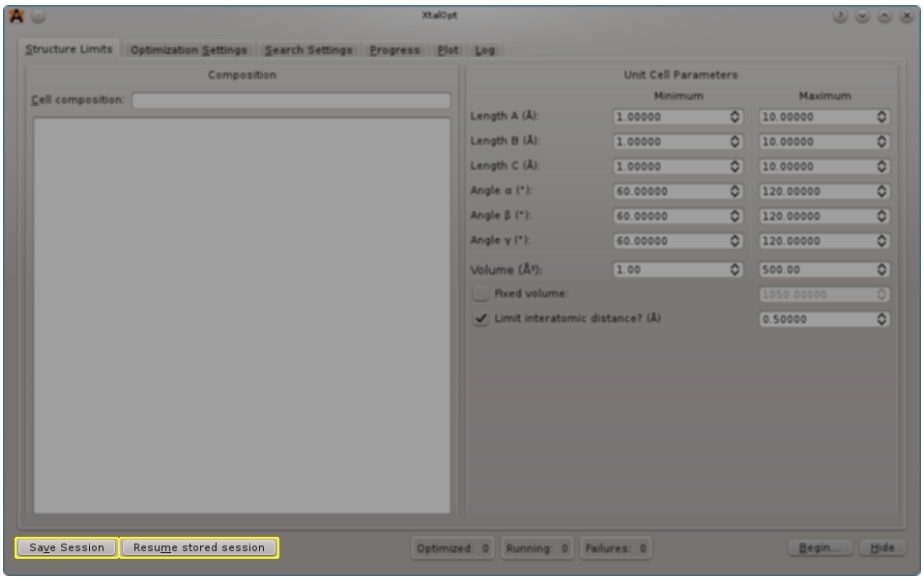
- Optimization Schemes
- Saving and Resuming Sessions in XtalOpt

Chapter 3

Saving and Resuming Sessions in XtalOpt

3.1 Contents

- How to save your session
- How to resume your session



3.2 How to save your session

XtalOpt will write a small file named xtalopt.state to its working directory that contains the information necessary to resume the session at a later time. The file can be rewritten manually by clicking the "Save Session" button highlighted above, and XtalOpt will automatically save the session every time a structure is updated.

XtalOpt will also write a file "structure.state" in each candidate structure's directory. This file stores XtalOpt-specific information about the structure.

3.3 How to resume your session

To resume a session, simply click "Resume stored session" (highlighted above) and select the xtalopt.state file in the working directory of the session you would like to resume. XtalOpt will then begin to load the structures and search parameters. You can monitor the progress with the progress bar that appears at the bottom of the window.

While the structures are loading, you may encounter errors that say:

```
Error, no (or not appropriate for [OPTIMIZER]) xtal data in [DIRECTORY].

This could be a result of resuming a structure that has not yet done
any local optimizations. If so, safely ignore this message.
```

As mentioned in the message, these can typically be ignored if it only happens for a handful of structures. This occurs when a structure has been generated in XtalOpt, but it has not completed any geometry optimization so there are no output files from which to load the geometry. If it happens for a significant number of structures (or structures that are known to have completed at least one geometry optimization step), the output files from the optimizer may be missing or corrupt.

After resuming a session, XtalOpt will ask if you would like to continue the search or enter read-only mode. Read-only mode will not generate new structures or submit geometry optimizations.

Note

If you are considering resuming a read-only session, take a look at the results.txt file in the working directory. It contains a list of all structures, sorted by enthalpy, with additional useful information. This can save some time when trying to locate the most stable structure of a old search.

The working directories for XtalOpt are relocatable, meaning that the directory containing xtalopt.state and the [gen]x[id] structure folders may be moved, tarred, zipped, etc. and still be resumed at a later time from a different location on the filesystem, or even a different computer.

Chapter 4

Optimization Schemes

4.1 Contents

- Overview: What are optimimization schemes, and why use them?
  - In a nutshell..
  - More details
- Optimization scheme user interface
  - Optimization step list
  - Add new optimization step
  - Remove current optimization step
  - Select template
  - Template editor
  - Save scheme
  - Resume scheme
- How to build an optimization scheme?
- How to save an optimization scheme for later?
- How to load an optimization scheme?
- What is saved?
- Suggestions for optimization schemes
  - Crystals (XtalOpt)

4.2 Overview: What are optimimization schemes, and why use them?

4.2.1 In a nutshell..

An optimization scheme is a series of optimization steps ("optsteps") that are to be performed in sequence on a structure. Each optimization step consists of a set of input file templates for the queuing system and optimizer to be used, and the structure is updated after each completes. So if an optimization scheme contains three optimization steps, a stucture's lifecycle is:

1. Generation of initial structure
2. Perform optstep 1 on initial structure
3. Update structure from the results of optstep 1
4. Perform optstep 2 on current structure (result of optstep 1)
5. Update structure from the results of optstep 2
6. Perform optstep 3 on current structure (result of optstep 2)
7. Update structure from the results of optstep 3
8. Current structure (result of optstep 3) is either accepted into the breeding pool or discarded, depending on its enthalpy relative to the other optimized structures.

4.2.2 More details

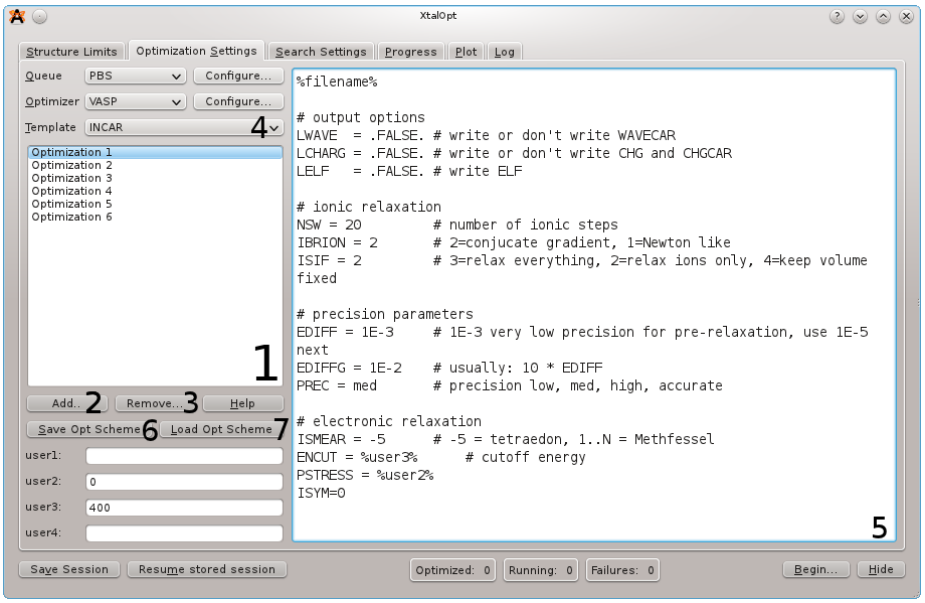
The efficiency of searching a potential energy surface for a global minimum can be significantly improved by moving each candidate structure to the nearest local minimum, i.e. performing a geometry optimization. The differences between searching with and without carrying out these local optimizations are explored in detail in Woodley SM, Catlow CRA. Comp. Mat. Sci. 2009;45(1):84-95 (Available at: http://linkinghub.elsevier.com/retri
).

Why not just perform a single geometry optimization on each structure? Stochastic search techniques, such as XtalOpt, will often need to perform geometry optimizations on structures that are far from a stationary point on the potential energy surface. For example, the randomly generated structures in the first generation of an evolutionary search are often highly disordered with unrealistic atomic separations. If these structures were to be optimized in a single step with accurately small convergence criteria, it would be quite expensive. Also, it is more than likely that most of the optimizations would not finish successfully before reaching the maximum number of geometry steps allowed by the optimizer or specified in the input. A second issue is that complex structures (periodic crystals, for example) often have so many degrees of freedom that convergence in a single step is difficult from a poor starting point (consider the effect on atomic coordinates when a unit cell's translation vector is modified).

The first problem (effectively optimizing to small convergence) can be solved by implementing an optimization scheme that optimizes to successively smaller convergence cutoffs.

The second problem can be addressed by reducing the degrees of freedom in the early optsteps and only optimizing everything once each component has individually converged to a reasonable parameterization. See Suggestions for optimization schemes for examples.

4.3 Optimization scheme user interface



We will use the above screenshot as we describe the process of creating, saving, and loading optimization schemes. The numbers indicate:

1. List of optimization steps
2. Button to add new optimization step
3. Button to remove current optimization step
4. Template selection menu
5. Template editor
6. Button to save current optimization scheme to file
7. Button to load optimization scheme from file

4.3.1 Optimization step list

This list shows the currently available optimization steps in the order that they will be performed. The optstep that is currently selected for editing is highlighted, and the editable optstep can be selected by clicking the appropriate entry.

4.3.2 Add new optimization step

Clicking this button will append a new optimization step to the optstep list. The new optstep's templates will be copies of the currently selected optstep's templates.

4.3.3 Remove current optimization step

Click this button to delete the currently selected optimization step.

4.3.4 Select template

This menu contains the filenames of the templates that are required by the currently selected queuing system (e.g. PBS, SGE, local...) and optimizer. The currently selected template is displayed in the template editor, and selecting a different template will update the editor.

4.3.5 Template editor

This text editor is used to view and edit the currently selected template for the current optstep.

4.3.6 Save scheme

This button will prompt for a location to save a .scheme file containing the current optimization step.

4.3.7 Resume scheme

This button will prompt for an existing .scheme file to load.

4.4 How to build an optimization scheme?

Creating a working scheme from scratch may take some time. We recommend checking the samples/ directory of the source code to obtain sample scheme for each optimizer (see How to load an optimization scheme?) and verifying that they are appropriate for the system under consideration before starting a search.

If there is not an appropriate sample, the following prescription may be used to generate your own:

1. Generate a random structure of the system under consideration. This may be done by hand, or by running a search just long enough to create the first random generation and saving one of the structures.

2. Create a starting optstep with the desired convergence criteria
3. Manually submit the optimization
4. If the optimization fails:
   (a) First determine why -- if the maximum iterations were exceeded or the optimization was aborted due to a badly performing minimizer, try one of the ideas below. Other optimization problems are beyond the scope of this document.
   (b) Reduce the convergence criteria of the current trial optstep
   (c) Remove degrees of freedom, e.g. by fixing cell parameters, atomic positions, etc
   (d) Reduce the accuracy of the calculation in other ways (use a courser integration grid, etc).
   (e) Change the minimizer (e.g. tell the optimizer to use conjugate gradients rather than BFGS, etc)
5. Once the optimization succeeds, create another set of input files with the desired convergence criteria for all degrees of freedom.
6. Manually submit the new optimization step. If it fails, try the ideas above until it converges.
7. Once the structure has converged to the desired level of accuracy, try to optimize another randomly generated structure using the optsteps that succeeded previously. Refine them if needed.
8. Once you have successfully optimized enough random structures that you are confident in your method, gather all of the inputs used and write your scheme from them.

The scheme may be written by copying each input file into the template editor (with the appropriate optstep and template selected, of course) and replacing the structure-specific information with the appropriate keywords. Click the "Help" button for the complete list of keywords.

We have found that the optimization schemes are surprisingly transferable within an optimizer, so once you have a working optimization scheme for a given optimization code only minor tweaks (usually to the energy cutoffs, etc) are necessary to use it on a different chemical system.

4.5 How to save an optimization scheme for later?

Once you have written your optimization scheme, you will want to save it for fast retrieval later (otherwise you will need to copy/paste and edit all of the templates again!). To save, simply click the "Save Opt Scheme" button and enter an appropriate filename with an extension of .scheme.

4.6 How to load an optimization scheme?

Loading an optimization is quite simple -- just click the "Load Opt Scheme" button and select the .scheme file you wish to load. This will also update the current queuing system and optimizer to those specified by the scheme.

4.7 What is saved?

The optimization scheme files contain more than just the templates for each optstep. They also store queue and optimizer specific settings. This is useful for storing configuration options for different clusters along with the scheme. Note that although XtalOpt will prompt for an SSH password if needed, it is **NOT** stored in the scheme file.

4.8 Suggestions for optimization schemes

4.8.1 Crystals (XtalOpt)

The following list describes the optimization steps used in the samples/vasp-xtalopt.scheme file distributed with the XtalOpt source code:

1. Fix unit cell, only optimize atomic coordinates. A very loose convergence criterion is used, and the number of KPOINTs is kept small.
2. The cell volume is fixed, but atomic positions and cell parameters are allowed to vary. The convergence criteria is the same as before, as is the KPOINT grid.
3. All degrees of freedom are considered using the same convergence criteria as before, but with a finer KPOINT grid.
4. Same as before, but with a stricter convergence criteria.
5. Same as before, but with a stricter convergence criteria and more KPOINTs.
6. Same as before, but with more KPOINTs.

This is only one of many possible optimization schemes that may work for crystals. It may need to be modified to work for your particular system.